



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/668,110	09/22/2000	Mark E. Kriegsman	11125-014001	9580

26161 7590 07/16/2010  
FISH & RICHARDSON PC  
P.O. BOX 1022  
MINNEAPOLIS, MN 55440-1022

EXAMINER
----------

BAYARD, DJENANE M

ART UNIT	PAPER NUMBER
----------	--------------

2444

NOTIFICATION DATE	DELIVERY MODE
-------------------	---------------

07/16/2010

ELECTRONIC

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

PATDOCTC@fr.com

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE BOARD OF PATENT APPEALS  
AND INTERFERENCES

---

*Ex parte* MARK E. KRIEGSMAN and BENJAMIN W. WYCKOFF

---

Appeal 2009-011064  
Application 09/668,110  
Technology Center 2400

---

Before JOHN A. JEFFERY, HOWARD B. BLANKENSHIP, and  
THU A. DANG, *Administrative Patent Judges*.

JEFFERY, *Administrative Patent Judge*.

DECISION ON APPEAL<sup>1</sup>

Appellants appeal under 35 U.S.C. § 134(a) from the Examiner's rejection of claims 1-3, 5-17, 19-21, and 23-28. We have jurisdiction under 35 U.S.C. § 6(b), and we heard the appeal on July 13, 2010. We affirm-in-part.

---

<sup>1</sup> The two-month time period for filing an appeal or commencing a civil action, as recited in 37 C.F.R. § 1.304, or for filing a request for rehearing, as recited in 37 C.F.R. § 41.52, begins to run from the "MAIL DATE" (paper delivery mode) or the "NOTIFICATION DATE" (electronic delivery mode) shown on the PTOL-90A cover letter attached to this decision.

## STATEMENT OF THE CASE

Appellants invention serves dynamic web pages by updating obsolete parts of those pages. *See generally* Abstract; Spec. 2-3. Claim 1 is illustrative with a key disputed limitation emphasized:

1. A method for enabling the generation of an updated web-page for storage in one of a plurality of cache servers, said method comprising:

implementing programmable rules executing on each of the plurality of cache servers, each programmable rule defining a triggering event associated with its corresponding cache server, the occurrence of the triggering event being *indicative of the existence of an obsolete portion of said web-page stored in said corresponding cache server*;

detecting an occurrence of a triggering event at a particular cache server selected from the plurality of cache servers;

in response to the occurrence of said triggering event, causing said particular cache server to request an update of said obsolete portion; and

receiving an updated portion of said web-page for storage at said particular cache server.

The Examiner relies on the following as evidence of unpatentability:

Leshem	US 5,870,559	Feb. 9, 1999
Kung	US 5,933,837	Aug. 3, 1999
Scherr	US 6,799,248 B2	Sept. 28, 2004 (eff. filed June 6, 1996)

### THE REJECTIONS

1. The Examiner rejected claims 1-3, 5, 8-16, 19-21, 23, and 26-28<sup>2</sup> under 35 U.S.C. § 102(e) as anticipated by Scherr. Ans. 3-6.<sup>3</sup>
2. The Examiner rejected claims 6 and 24 under 35 U.S.C. § 103(a) as unpatentable over Scherr and Kung. Ans. 7.
3. The Examiner rejected claims 7, 17, and 25 under 35 U.S.C. § 103(a) as unpatentable over Scherr and Leshem. Ans. 7-8.

### CLAIM GROUPING

Appellants argue the following claim groupings separately regarding the anticipation rejection: (1) claims 1, 13, and 19; (2) claims 3 and 21; (3) claims 5 and 23; (4) claims 8 and 26; and (5) claim 16. *See* App. Br. 6-12. Accordingly, we treat each group separately, and select claims 1, 3, and 8 as representative of groups (1), (2), and (4), respectively. *See* 37 C.F.R. § 41.37(c)(1)(vii).

### THE ANTICIPATION REJECTION

Regarding representative claim 1, the Examiner finds that Scherr updates web pages stored in a cache server where a “triggering event” (a predetermined time period or a user logon) indicates the existence of an obsolete part of the web page for updating. Ans. 3-4, 8-9. Appellants argue

---

<sup>2</sup> Although the Examiner includes cancelled claims 4 and 22 in this rejection (Ans. 3-4), we deem this error harmless and present the correct claim listing here for clarity.

<sup>3</sup> Throughout this opinion, we refer to (1) the Appeal Brief filed July 15, 2008 (supplemented August 8, 2008); (2) the Examiner’s Answer mailed October 30, 2008; and (3) the Reply Brief filed December 29, 2008.

that while a lapse of a time interval and user logon are “triggering events,” they are not indicative of an obsolete part of a webpage as claimed. App. Br. 6-8; Reply Br. 2-5.

Appellants add that Scherr also does not (1) interpret a script as recited in claim 3 (App. Br. 8-9); (2) detect the triggering event by detecting the receipt of an updated part of the webpage as recited in claim 5 (App. Br. 9-10; Reply Br. 5); (3) cause the cache server to request an update from an origin server, and receive an updated portion from the origin server as recited in claim 8 (App. Br. 11-12; Reply Br. 5-6); and (4) disclose a communication path between a programmable script and an administrator process enabling the script to receive instructions from the administrator process as recited in claim 16 (App. Br. 12; Reply Br. 6). The issues before us, then, are as follows:

### ISSUES

Under § 102, has the Examiner erred by finding that Scherr:

(1) discloses a triggering event indicative of the existence of an obsolete portion of a webpage stored in a cache server as recited in claim 1?

(2) interprets a script containing instructions for defining a rule as recited in claim 3?

(3) detects the triggering event by detecting the receipt of an updated part of the webpage as recited in claim 5?

(4) causes a cache server to request an update from an origin server, and receive an updated portion from the origin server as recited in claim 8?

(5) discloses a communication path between a programmable script and an administrator process enabling the script to receive instructions from the administrator process as recited in claim 16?

### FINDINGS OF FACT (FF)

1. To improve perceived response time, Scherr's cache management system 10 can use different methods involving page caching or data usage frequency. For example, after a request for data, the system can fetch the data from the network using a time-sensitive cache management method depending on whether its time stamp is within certain parameters. Also, a local site 06 may use a time-currency cache management method to refresh pages at various frequencies (e.g., every 15 minutes during trading hours for a given stock exchange, and another frequency when the exchange is closed). Scherr, Abstract; col. 5, ll. 49-61; col. 6, ll. 5-24; col. 8, ll. 44-58; col. 8, l. 66 – col. 9, l. 23; Figs. 1a, 2a, 4.

2. Figure 12 shows scripted variables and pseudo-code for time-sensitive cache management. Scripted parameters CC indicate that pages are to be kept fresh during the trading hours of a stock exchange (i.e., between 1000 and 1600 hours). During that time, the pages are refreshed every 15 minutes ("VALUE1" = 15). Scherr, col. 9, ll. 10-22; Fig. 12.

3. A variation of the time-sensitive method can include a request that nothing cached be out of date more than some specified time period. Scherr, col. 9, 20-22.

4. Time-sensitive cache management can involve charging users for the "freshness" of fetched web pages (e.g., charge a premium to ensure that pages are less than 7 hours old). Scherr, col. 10, ll. 51-56.

5. When users first log on, the local site's cache management system 10 can pre-fetch web pages from requested sites. This technique can be used in conjunction with tracked data usage patterns. For example, if users of a financial journal habitually go to a stock quote site after finishing the journal pages, the stock quote pages can be pre-fetched every time the journal pages are fetched. Scherr, col. 6, ll. 27-40.

6. In an alternative embodiment, another form of indexing or pre-fetching involves locally mirroring a site. Also, service providers could supply the mirror files by sending unsolicited updates to a local site as they occur. Scherr, col. 11, ll. 38-57.

7. The administrator of backbone link 04 can configure its cache management system to use page usage or data usage patterns. All of these administrative decisions and actions can be performed by an expert system dynamically. Scherr, col. 5, ll. 62-67; Fig. 1a.

7A. The configurator acts as a cache memory manager and establishes parameters and other indicators that may be needed by the cache management method(s) selected by the site. Scherr, col. 6, ll. 57-65; Figs. 1c, 2a (steps 22-24).

## PRINCIPLES OF LAW

To anticipate under § 102, the prior art reference “must not only disclose all elements of the claim within the four corners of the document, but must also disclose those elements arranged as in the ‘claim.’” *Net MoneyIn, Inc. v. Verisign, Inc.*, 545 F.3d 1359, 1369 (Fed. Cir. 2008) (citation and internal quotation marks omitted).

“Thus, it is not enough that the prior art reference discloses part of the claimed invention, which an ordinary artisan might supplement to make the whole, or that it includes multiple, distinct teachings that the artisan might somehow combine to achieve the claimed invention.” *Id.* at 1371. *See also In re Arkley*, 455 F.2d 586, 587 (CCPA 1972) (“[T]he [prior art] reference must clearly and unequivocally disclose the claimed [invention] or direct those skilled in the art to the [invention] without *any* need for picking, choosing, and combining various disclosures not directly related to each other by the teachings of the cited reference.”).

## ANALYSIS

### *Claims 1, 2, 11-15, 19, 20, 27, and 28*

We begin by construing the key disputed limitation of claim 1 which calls for, in pertinent part, a “triggering event” indicative of the existence of an *obsolete* part of a webpage. The key word here is “obsolete,” for Appellants do not dispute that Scherr’s time-based and logon-based cache management methods cited by the Examiner are “triggering events,”<sup>4</sup> but rather that these events have nothing to do with whether the pages are obsolete or not. App. Br. 6-8; Reply Br. 2-5.

But despite these contentions, Appellants never actually squarely define what “obsolete” means, either in the Specification or in the Briefs. We therefore construe the term “obsolete” with its plain meaning (i.e., the

---

<sup>4</sup> *See, e.g.,* Reply Br. 3 (“Applicant agrees that the lapse of a time interval is indeed a known ‘*triggering event*.’”); *see also* App. Br. 8 (“[T]he *triggering event* ‘a user has logged on’ cannot possibly be regarded as ‘indicative of the existence of an obsolete portion’ of any web page.”) (emphases added).



ordinary and customary meaning given to the term by those of ordinary skill in the art). *See Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005) (en banc).

The term “obsolete” means “[n]o longer in use; gone into disuse; disused; neglected; as, an obsolete word; an obsolete statute . . . .”<sup>5</sup> Based on this definition, we find no error in the Examiner’s reliance on Scherr’s time-based cache management method which refreshes pages every 15 minutes. FF 1-2.

We reach this conclusion emphasizing that the term “obsolete” is not only broad, but relative.<sup>6</sup> As such, nothing in the claim precludes an older page that is refreshed every 15 minutes from being an “obsolete” page (i.e., a page no longer in use), at least with respect to the newer, refreshed version of the page. Notably, this “obsolescence” would be particularly acute in Scherr’s stock-market context—an environment in which information rapidly changes. *See id.* In any event, that Scherr can condition requests to ensure that data is not *out of date* (FF 3) or charge users for fetching “*fresh*” web pages (e.g., less than 7 hours old) (FF 4) only bolsters our conclusion that Scherr discloses a triggering event indicative of the existence of an obsolete part of a web page to ensure that these temporal conditions are met.

We reach a similar conclusion regarding the user-logon “triggering event,” for it too would result in fetching fresh versions of pages every time the user logs on or habitually uses data in a particular manner. *See* FF 5.

---

<sup>5</sup> Webster's Revised Unabridged Dictionary, 1913 ed., *available at* <http://machaut.uchicago.edu/?resource=Webster%27s&word=obsolete&use1913=on>.

<sup>6</sup> *Accord* Spec. 5:17-18 (noting that “obsolescence can strike an object at any time . . . .”).

Although the pages are pre-fetched responsive to this triggering event, they would nevertheless update earlier, “obsolete” pages (e.g., stock quote pages from previous sessions). *See id.*

We are therefore not persuaded that the Examiner erred in rejecting representative claim 1, and claims 2, 11-15, 19, 20, 27, and 28 not separately argued.

### *Claims 3 and 21*

We likewise sustain the Examiner’s rejection of representative claim 3. As the Examiner indicates (Ans. 10), Scherr’s script in Figure 12 updates a web page every 15 minutes (FF 2)—a script that is interpreted to define a triggering event indicative of the existence of an “obsolete” part of a web page, namely the older page that is refreshed as noted above.

We are therefore not persuaded that the Examiner erred in rejecting representative claim 3, and claim 21 not separately argued.

### *Claims 5 and 23*

We will not, however, sustain the Examiner’s rejection of claim 5. Although the Examiner is correct that the unsolicited updates sent by Scherr’s service providers to local sites for mirroring purposes would be detected (Ans. 10; FF 6), we fail to see how this detection associated with mirroring would also detect the *triggering event* as claimed—a separate and distinct event that is associated with a predetermined time period or user logon as noted above. *Compare* FF 1-5 *with* FF 6. Moreover, claim 1 (from which claim 5 depends) requires the server to request an update of the

obsolete portion *responsive to* this triggering event—a “triggered” request that would be rendered meaningless by the automatic updates provided via the mirroring feature. *See* FF 6.

In any event, that Scherr’s mirroring feature is an *alternative embodiment* only bolsters our conclusion that this detection does not necessarily detect the temporal and user-logon triggering events noted above. In short, the Examiner’s combination of features from different embodiments runs afoul of the law of anticipation which requires that the cited reference discloses each limitation *arranged as in the claim*—which it does not. *See Net MoneyIn*, 545 F.3d at 1369; *see also Arkley*, 455 F.2d at 587.

We are therefore persuaded that the Examiner erred in rejecting claim 5, and claim 23 which recites commensurate limitations.

#### *Claims 8-10 and 26*

We will, however, sustain the Examiner’s rejection of representative claim 8. As the Examiner indicates (Ans. 11), Scherr fetches data *from the network* to update pages (FF 1)—a network-based service that fully meets an “origin server” as claimed. Although Scherr does not use the term “server” in this context or otherwise detail the source of this network-based data, this data nevertheless originates from the network, and therefore must involve some type of server to provide the data via that network. *See id.*

We are therefore not persuaded that the Examiner erred in rejecting representative claim 8, and claims 9, 10, and 26 not separately argued.

*Claim 16*

We will also sustain the Examiner's rejection of claim 16. Although Scherr does not detail the communication link associated with the capability of the administrator of backbone link to configure the cache management system (which can be an expert system) (FF 7), the configurator nevertheless performs administrator functions by establishing parameters and other indicators needed by the selected cache management methods (FF 7A). To achieve this end, there must be some sort of communication path between the script and this administrator process to enable the script to receive instructions (e.g., parameters) from the administrator process. *See* FF 2, 7A.

We are therefore not persuaded that the Examiner erred in rejecting claim 16.

THE OBVIOUSNESS REJECTION OVER SCHERR AND KUNG

Regarding representative claim 6, the Examiner finds that Scherr discloses every claimed feature except for requesting an update by formulating a database query, and cites Kung for this teaching in concluding the claim would have been obvious. Ans. 7, 12. Appellants argue that (1) the Examiner's reliance on Kung is flawed since Kung's primary system does not teach requesting updates as claimed, but rather provides updates to subscribing systems (App. Br. 12-14; Reply Br. 6-7), and (2) there is no reason to combine the references (App. Br. 14-15; Reply Br. 7-8). The issues before us, then, are as follows:

## ISSUES

(1) Under § 103, has the Examiner erred in rejecting claim 6 by finding that Scherr and Kung collectively would have taught or suggested causing a cache server to request an update of the obsolete portion of a web page by formulating a database query to be carried out by a database engine?

(2) Is the Examiner's reason to combine the teachings of these references supported by articulated reasoning with some rational underpinning to justify the Examiner's obviousness conclusion?

## ADDITIONAL FINDINGS OF FACT

8. Kung's system synchronizes replicated data across heterogeneous databases in a network. Specifically, a primary system 102 includes primary database 112 and data consistency server 114 that is connected to subscribing systems 104 via network 106. The subscribing systems include database engines 116, 118, and 120 which receive formatted database updates and updates databases 112, 124, 126. Kung, Abstract; col. 1, ll. 7-10; col. 3, ll. 12-37; Fig. 1.

9. A query manager 204 connected to the primary database engine (1) generates queries only for the databases that need updates, and (2) translates queries based on a specified format for each of the subscribing databases. The translated queries are then distributed to the subscribing databases. Kung, col. 2, ll. 38-57; col. 3, ll. 38-62; col. 5, ll. 1-12; Figs. 2, 3.

## ANALYSIS

We will sustain the Examiner's obviousness rejection of claim 6. Although Kung's primary system provides updates to secondary systems by formulating database queries (FF 8-9), the Examiner's rejection is not based on Kung alone, but rather the collective teachings of Scherr and Kung. *See* Ans. 7, 12. While we agree with Appellants (Reply Br. 6) that claim 6 requires performing the update by a cache server—not an origin server—we nonetheless see no reason why Kung's teaching of using database queries in connection with updating data (FF 8-9) could not have been applied to Scherr's cache management system which requests updates to pages responsive to various triggering events as noted previously. *See* FF 1-5. In short, formulating database queries to that end is tantamount to the predictable use of prior art elements according to their established functions. *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 417 (2007). We therefore find the Examiner's reason to combine the teachings of the cited references supported by articulated reasoning with some rational underpinning to justify the Examiner's obviousness conclusion.

We are therefore not persuaded that the Examiner erred in rejecting representative claim 6, and claim 24 which falls therewith.

## THE OBVIOUSNESS REJECTION OVER SCHERR AND LESHEM

Regarding claim 7, the Examiner finds that Scherr discloses every recited feature except for an assembly script for assembling constituent and updated portions into an updated web page, and cites Leshem for this feature in concluding the claim would have been obvious. Ans. 7-8. Appellants

argue that Leshem's creating site maps by automatically filling out forms does not teach or suggest the recited assembly script (App. Br. 15-17). The issue before us, then, is as follows:

### ISSUE

Under § 103, has the Examiner erred in rejecting claim 7 by finding that Scherr and Leshem collectively would have taught or suggested an assembly script for assembling constituent and updated portions into an updated web page?

### ADDITIONAL FINDINGS OF FACT

10. Leshem discloses a web site analysis program that scans and maps dynamically-generated web pages responsive to queries by (1) capturing and recording a dataset manually entered by the user into an embedded form of a web page, and (2) automatically resubmitting the dataset when the web site is re-scanned. Leshem, Abstract; col. 3, ll. 30-43.

11. During the dataset capture session, any pages retrieved by the browser and any forms (datasets) submitted by the browser are automatically recorded by the web site analysis program into the site map. When the site map is later updated, the scanning routines automatically re-enter the captured datasets into the corresponding forms and recreate the form submissions. The dynamically-generated web pages returned responsive to these submissions are then added to the updated site map as nodes. Leshem, col. 3, ll. 44-63.

## ANALYSIS

We will not sustain the Examiner's obviousness rejection of claim 7. First, it is unclear exactly how the Examiner is equating Leshem's site map to the recited assembly script with assemblies constituent and updated portions into a web page. Apart from merely reiterating the claim language and citing Leshem's background discussion of this feature as alleged support for these limitations (Ans. 8), the Examiner does not explain how or why this site *map* would have taught or suggested the assembly script limitation that creates updated web *pages*.

In any event, as Appellants indicate (App. Br. 15-17), Leshem pertains to a web site analysis program that *maps* web pages that are dynamically-generated responsive to form submissions. FF 10-11. Although these dynamically-generated pages are added to an updated site map as nodes (FF 11), we fail to see how this process assembles both constituent and updated portions into an updated web page as claimed.<sup>7</sup> Therefore, even if the references were combinable, they still would not teach or suggest the recited assembly script limitation of claim 7 and claim 25 which recites commensurate limitations. Since this issue is dispositive of our reversal of claim 7, we need not address Appellants' other arguments regarding the impropriety of combining the references (App. Br. 17-18; Reply Br. 8).

---

<sup>7</sup> Notably, the Examiner failed to respond to Appellants' challenge in this regard in the Answer, but instead noted that the rejection did not amount to impermissible hindsight reconstruction of the invention. See Ans. 13.



We will, however, sustain the Examiner's rejection of claim 17 which, unlike claims 7 and 25, does not require assembling *both* constituent and updated portions into a web page, but rather recites *only* assembling constituent portions into a web page. Since Scherr's page generation capabilities amply teach this feature (*see* FF 1-5), we find the Examiner's reliance on Leshem for this feature merely cumulative. We therefore need not address Appellants' arguments regarding the impropriety of combining the references (App. Br. 17-18; Reply Br. 8).

We are therefore persuaded that the Examiner erred in rejecting claims 7 and 25, but are not persuaded the Examiner erred in rejecting claim 17.

### CONCLUSION

Under § 102, the Examiner did not err in rejecting claims 1-3 and 8-16, but erred in rejecting claims 5 and 23. Under § 103, the Examiner did not err in rejecting claims 6, 17, and 24, but erred in rejecting claims 7 and 25.

### ORDER

The Examiner's decision rejecting claims 1-3, 5-17, 19-21, and 23-28 is affirmed-in-part.

Appeal 2009-011064  
Application 09/668,110

No time period for taking any subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a)(1)(iv).

AFFIRMED-IN-PART

pgc

FISH & RICHARDSON PC  
P.O. BOX 1022  
MINNEAPOLIS, MN 55440-1022